

Lezione 4 – Costrutti di selezione e iterazione

Programmazione

Lab di Programmazione

Costrutti di selezione e iterazione

Filippo Gaudenzi

Università degli Studi di Milano

Esercizi di oggi

1. Esercizio: si scriva un programma che chiede all'utente di indicare una cifra da investire e calcola l'entità delle commissioni richieste per l'investimento.
2. Esercizio: si scriva un programma che fornisca moltiplicazione e divisione attraverso le operazioni di somma e sottrazione
3. Esercizio: si scriva un programma che preso in input un numero binario di 8 cifre restituisca il valore in base 10
4. Esercizio: si scriva un programma che calcoli la serie di fibonacci (no ricorsione)

Statement if

CODE EXAMPLE

```
if (expression)
{
    Block of statements;
}
else
{
    Block of statements;
}
```

Expression: la condizione inserita nell'if deve restituire **un valore booleano (true - 1, false - 0)** è possibile unire più sottocondizioni attraverso gli operatori booleani

! NOT, && AND, || OR,

Statement do-while/while(1)

DO-WHILE

CODE EXAMPLE

```
do{  
    Block of statements;  
}while(expression);
```

Il ciclo è ripetuto **finchè l'espressione nel while è VERA.**

Il do-while viene eseguito **SEMPRE almeno una volta**

WHILE

CODE EXAMPLE

```
while(expression)  
{  
    Block of statements;  
}
```

Il ciclo è **ripetuto finchè l'espressione nel while è VERA.**

Il controllo avviene prima del ciclo, quindi il blocco potrebbe non essere mai eseguito

Statement do-while/while (2)

Il ciclo do-while viene spesso utilizzato per il controllo degli input

```
int n;  
do{  
    n=LeggeIntero();  
}while(n<0);
```

Il ciclo while viene utilizzato maggiormente nei cicli di iterazione dove prima viene effettuato il controllo se o no eseguire i comandi.

```
int n=3  
while(n>0){  
    n--;  
}
```

Ex 1 (1)

Affrontiamo il primo problema passo passo.

Definiamo astrattamente il problema

La commissione è data da una percentuale sul valore dell'investimento più un costo fisso. Parte variabile e costo fisso dipendono dagli scaglioni di investimento.

*commission=investment*cost_var+cost_fix;*

Identifichiamo il ruolo delle variabili

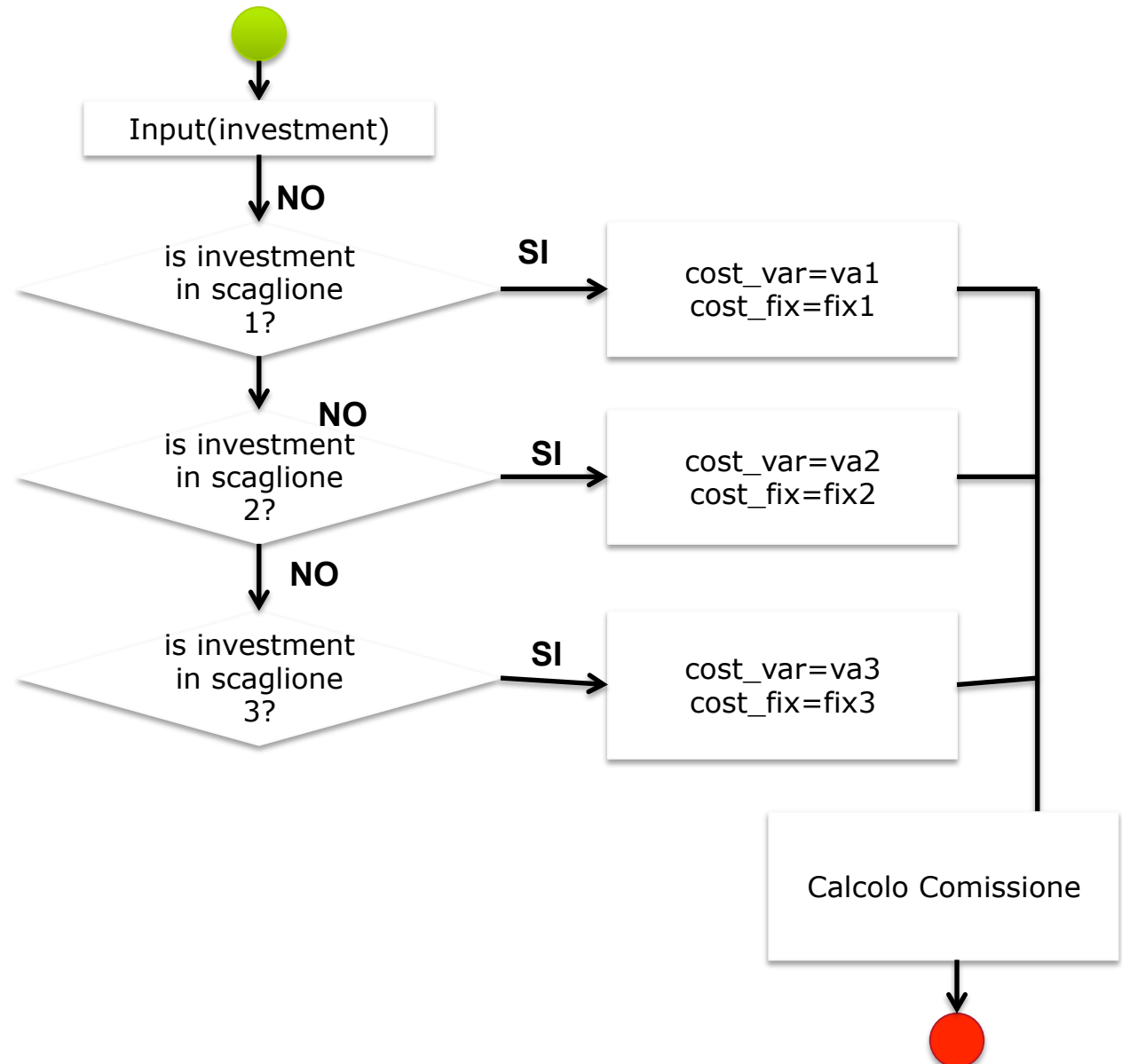
commission è evidentemente l'output

investment è l'input dell'utente

cost_var e cost_fix sono parametri del programma come lo sono gli scaglioni

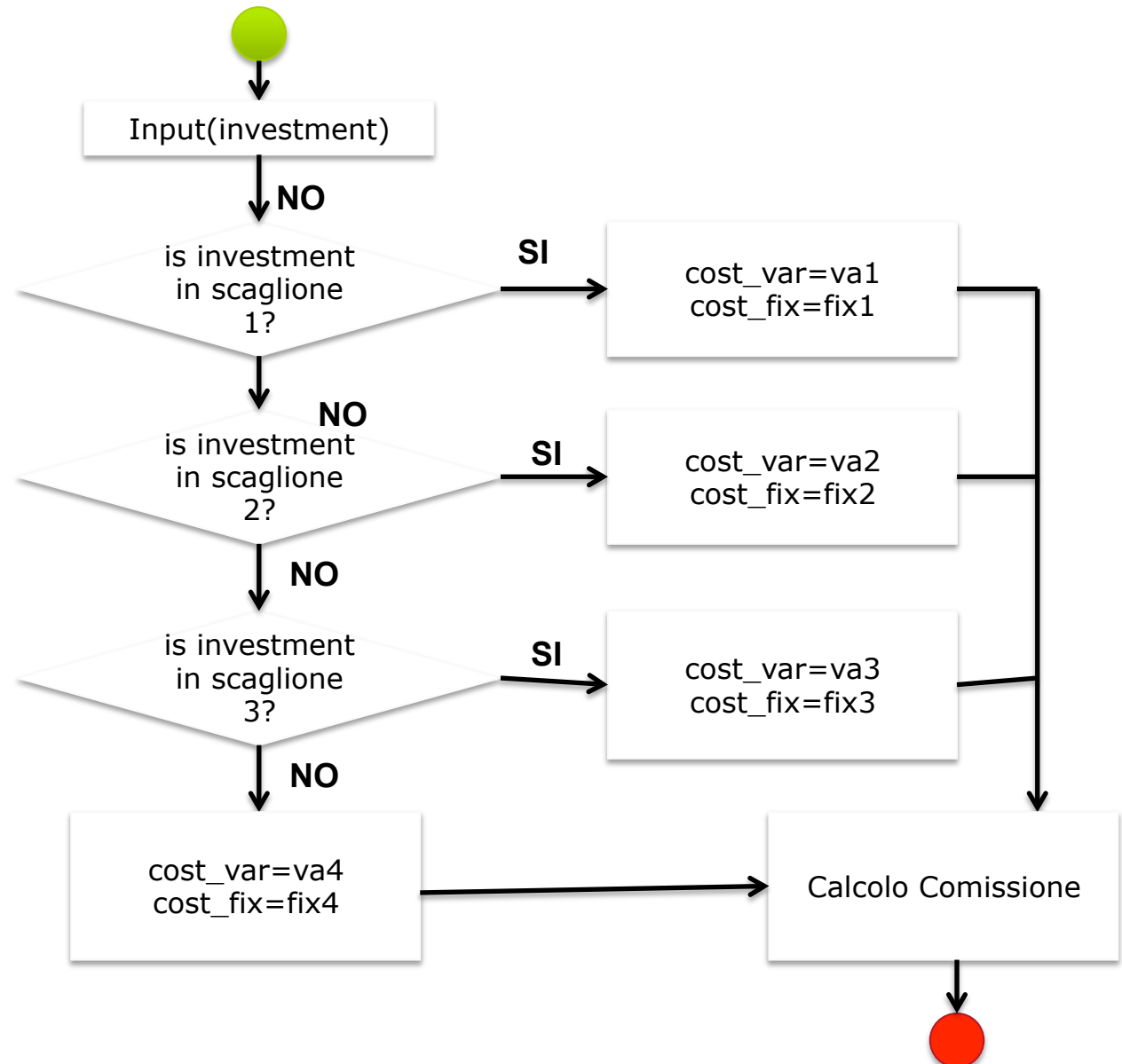
Ex 1 (2)

Definiamo il diagramma di flusso del programma

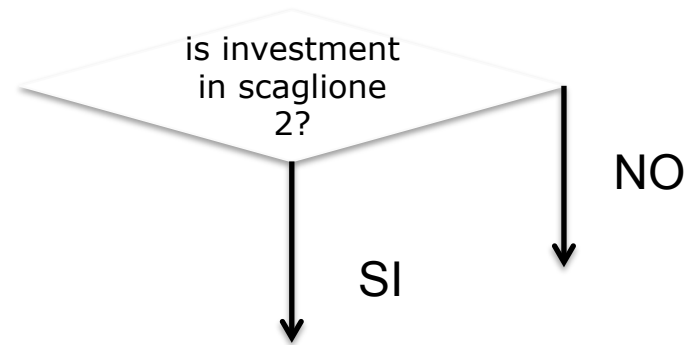


Ex 1 (2)

Definiamo il diagramma di flusso del programma

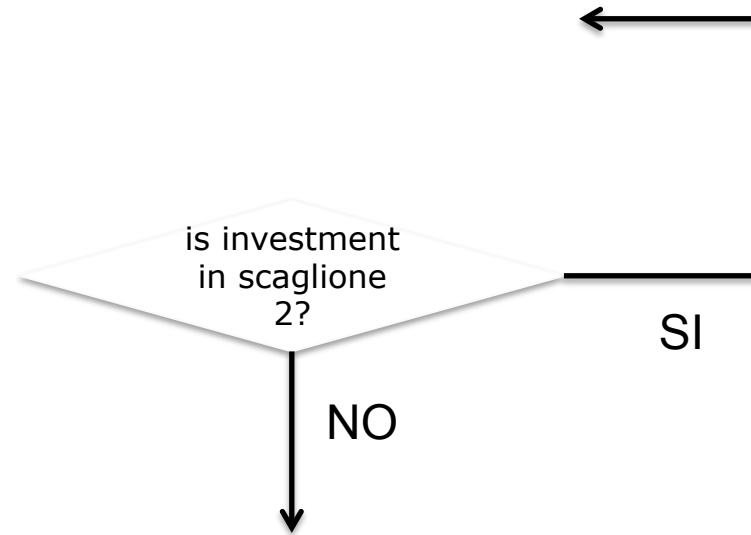


Traduzione Diagramma di flusso (IF)



Se entrambe le uscite proseguono verso la fine del programma allora stiamo utilizzando un if

Traduzione Diagramma di flusso (CICLO)



Se una delle uscite ritorna indietro nel diagramma di flusso la trasformazione è in un ciclo do-while/while/for?

Ex per casa

1. Si scriva un programma che prenda in input un carattere stampabile C e un numero intero N. Stampi per N righe il carattere C tale che alla riga 1 ci sia solo un carattere C, alla riga 2 due caratteri C e alla riga N, N caratteri C
2. Si scriva un programma che prenda in input una lettera dell'alfabeto A e un numero intero positivo N. Si stampi dalla lettera acquisita A tutte le N lettere successive dell'alfabeto. Se si arriva alla 'z' ricominciare dalla 'a'
3. Si scriva un programma che implementi il cifrario di cesare